

# My CRUD functions

Snel een tabel aanmaken en enkele formulieren hiervoor maken kan een tijdrovende bezigheid zijn. Gelukkig kunnen enkele functies ons het leven gemakkelijk maken. Telkens wij een tabel in een databank aanmaken wensen wij dezelfde functionaliteiten:

- Gegevens toevoegen = Create
- Gegevens lezen = Read
- Gegevens aanpassen = Update
- Gegevens verwijderen = Delete



Deze functies noemen we kortweg CRUD. Voor het maken van PHP pagina's die automatisch enkele pagina's voor CRUD maken zijn er genoeg websites te vinden (bv. <http://www.phpsc scaffold.com/>). Toch zorgt dit in mijn ogen niet voor een pasklaar resultaat. Meestal mis ik hier:

- Validatie van de formulieren
- Relaties tussen tabellen
- De gegevenstypen ENUM() en SET()
- Kalender voor data te kiezen bij gegevenstype DATE



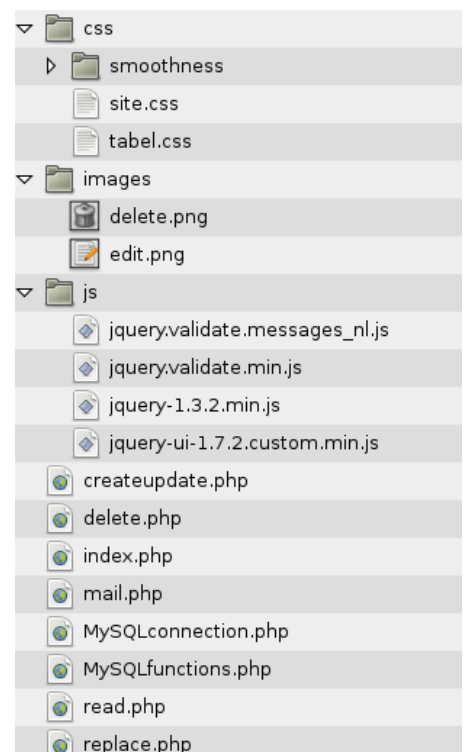
Daarom wil ik u zelf aan de slag laten gaan om uw eigen CRUD functies te maken in PHP, dit voor het benaderen van een MySQL databank.

Voor het valideren van het formulier heb ik in de functies gebruik gemaakt van JQuery (<http://jquery.com/>). Het toevoegen van een kalender aan het gegevenstype DATE was mogelijk met JQuery UI (<http://jqueryui.com/>).

Het toevoegen van relaties tussen tabellen doen we normaal gezien via foreign keys in de engine InnoDB. Maar vele online hosting providers bieden geen InnoDB aan maar enkel MyISAM. Het opzoeken van de relatie gaan we daarom expliciet in de opmerking van het veld noteren. Hierdoor kan het script de relaties verwerken. (Een oplossing voor het ontbreken van ON UPDATE of ON DELETE CASCADE in MyISAM is hier niet voorzien maar via deze functies snel toe te voegen.)

Hieronder ziet u een overzicht van de bestandenlijst en functies die we gemaakt hebben en gaan gebruiken.

```
3 class MySQLfunctions {
4
5 public function MySQLconnect($host,$user,$pass,$db) {
21
22 public function MySQLtable($db,$tabel,$fields,$pri key,$add delete = false,$add ec
158
159 public function MySQLform($tabel,$fields,$pri key,$id,$db,$exceptions=array()) {
364
365 public function MySQLdelete($tabel,$pri key,$id,$delete=false) {
392
393 public function MySQLdisconnect() {
403
404 private function GetGegevenstypes($tabel,$key) {
421
422 private function MySQLrequired($tabel,$key) {
444
445 private function GetFieldString($fields) {
463
464 public function MySQLreplace($tabel,$waarden) {
503
504 private function GetTypeNoParams($string) {
514
515 private function GetParamsFromType($string) {
526
527 private function GetTableCommentFromInformationSchema($db,$tabel,$column) {
544
545 }
```



Laten we starten met een index pagina te maken die ons naar de twee belangrijkste php pagina's zal brengen. De createupdate.php wordt zowel gebruikt voor het aanmaken van een nieuw record als voor het updaten. Indien geen id meegegeven (createupdate.php?id=1) dan zal een nieuw record toegevoegd worden.

#### index.php

```
<html>
<head>
<link type="text/css" href="css/site.css" rel="stylesheet" />
</head>
<body>
<h1>CRUD</h1><br/>
<a href="createupdate.php" target="_SELF">Create</a><br/>
<a href="read.php" target="_SELF">Read Update Delete</a><br/>
</body>
</html>
```

Eerst moeten we enkele variabelen aanmaken die de databankverbinding mogelijk maken. We starten met een nieuwe instantie van de mySQL functies aan te maken en de connectievariabelen juist te zetten. Vervolgens kiezen we de tabel en maken we een array van de velden uit de weer te geven tabel. De array is opgebouwd als "alias van het veld" => "veldnaam in MySQL". Optioneel kan u een WHERE, LIMIT of ORDER meegeven aan de SQL query.

#### MySQLconnections.php

```
<?php
// Class MySQLfunctions
$MyFunc= new MySQLfunctions;

// Calling the MySQLconnect function
$host="localhost";
$user="gebruiker";
$pass="paswoord";
$db="mydb";

$MyFunc->MySQLconnect($host,$user,$pass,$db);

//Selecting the table
$tabel="personeel";

//Array of alias and field
$fields= array("Voornaam" => "voornaam", "Naam" => "naam", "Opmerkingen" =>
"opmerkingen","Dienst" => "dienst", "Functies" => "functies", "Geboortedatum"
=> "geboortedatum");

//The Key?
$pri_key="id";

//SELECT WHERE?
$where = "";

//ORDER BY?
$order = null;

//LIMIT?
$limit = 7;
?>
```

De functie MySQLtable() maakt een HTM tabel van onze databank query. We kunnen enkele parameters meegeven:

- \$db = de databank (verplicht)
- \$tabel = de tabel (verplicht)
- \$fields = array van alias en velden (verplicht)
- \$pri\_key = de primaire sleutel (verplicht)
- \$add\_edit = boolean (true/false) voor het toevoegen van de knop aanpassen
- \$add\_delete = boolean (true/false) voor het toevoegen van de knop delete
- Optioneel kan je ook nog een \$where, \$order of \$limit toevoegen aan de sql query.

read.php

```
<html>
<body>

<link type="text/css" href="css/site.css" rel="stylesheet" />
<link type="text/css" href="css/tabel.css" rel="stylesheet" />
<script type="text/javascript" src="js/jquery-1.3.2.min.js"></script>
<script type="text/javascript" src="js/jquery.validate.min.js"></script>
<script type="text/javascript"
src="js/jquery.validate.messages_nl.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $("#MySQLtable tr").mouseover(
        function(){
            $(this).addClass("over");
        });
    $("#MySQLtable tr").mouseout(
        function(){
            $(this).removeClass("over");
        });

    $("#MySQLtable tr:even").addClass("alt");

    $("#MySQLnew").validate({
        submitHandler: function(form) {
            form.submit();
        }
    });
});
</script>

<h1>Overview records</h1>

<?php
include_once("MySQLfunctions.php");

include_once("MySQLconnection.php");

// Calling the MySQLtable function
$MyFunc->MySQLtable($db,$tabel,$fields,$pri_key,true,true,$where,$order,
$limit);
```

```
//Calling the MySQLsisconnect function
$MyFunc->MySQLdisconnect();

echo "<br/><br/><a href=\".\\" target=\"_SELF\">Ga terug...</a>";

?>

</body>
</html>
```

**Overview records**

Voornaam	Naam	Opmerkingen	Dienst	Funcities	Geboortedatum		
Bert	VdA	Niets speciaals te melden...	Marketing	Coördinator,Directie	2009-11-24		

[Ga terug...](#)

We vergelijken de weergave met de eigenlijke databank waarden. Merk op dat dienst en functies hier een id nummer heeft in plaats van de eigenlijke dienstnaam of functienaam.

```
mysql> SELECT * FROM personeel;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | voornaam | naam | opmerkingen | dienst | functies | timestamp | geboortedatum |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Bert | VdA | Niets speciaals te melden... | 2 | 2,1 | 2009-10-14 15:48:01 | 2009-11-24 |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0,00 sec)

mysql>
```

```
mysql> SELECT * FROM functies;
+----+-----+
| id | functie |
+----+-----+
| 3 | Administratie |
| 2 | Coördinator |
| 1 | Directie |
+----+-----+
3 rows in set (0,01 sec)

mysql>
```

```
mysql> SELECT * FROM diensten;
+----+-----+
| id | dienst |
+----+-----+
| 1 | Logistics |
| 2 | Marketing |
| 3 | Accounting |
| 4 | Sales |
+----+-----+
4 rows in set (0,01 sec)

mysql>
```

Het script gaat zelf op zoek naar de relaties indien deze in de opmerking van het tabelontwerp is meegegeven. Hieronder ziet u de databankontwerpen:

Table Name:  Comment:

Columns and Indices | Table Options | Advanced Options

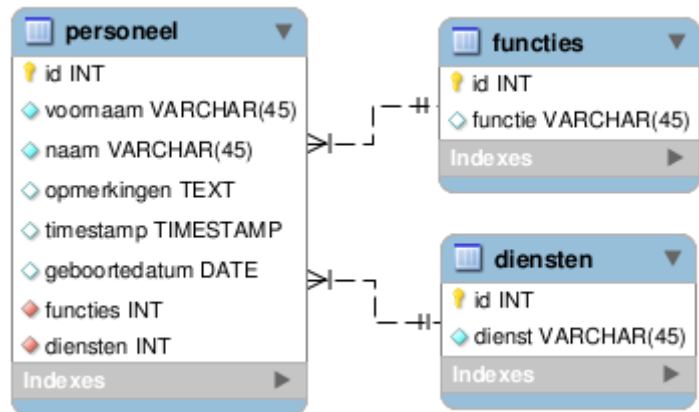
Column Name	Data Type	NOT NULL	AUTO INC	Flags	Default Value	Comments
id	INTEGER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		NULL	
voornaam	VARCHAR(25)	<input type="checkbox"/>	<input type="checkbox"/>		NULL	
naam	VARCHAR(25)	<input checked="" type="checkbox"/>	<input type="checkbox"/>		NULL	
opmerkingen	TEXT	<input type="checkbox"/>	<input type="checkbox"/>		NULL	
dienst	VARCHAR(25)	<input checked="" type="checkbox"/>	<input type="checkbox"/>		NULL	SELECT id, dienst FROM diensten
functies	TEXT	<input type="checkbox"/>	<input type="checkbox"/>		NULL	SELECT id, functie FROM functies
timestamp	TIMESTAMP	<input checked="" type="checkbox"/>	<input type="checkbox"/>		CURRENT_TIMESTAMP	
geboortedatum	DATE	<input type="checkbox"/>	<input type="checkbox"/>		NULL	

Het is belangrijk tijd te nemen voor het tabelontwerp, alle functionaliteit komt van dit ontwerp en zo hoort het ook!

Hieronder maken we een opsomming van de mogelijkheden alvorens verder te gaan naar het formulier.

### 1 Verplichte velden

Indien de optie NOT NULL aanwezig is zal het formulier dit valideren en indien leeg de gebruiker een melding geven.



### 2 Datum selecteren

Indien het gegevenstype DATE meegegeven wordt zal een kalender in het formulier aanwezig zijn.

### 3 Textveld of tekstregel

Indien het gegevenstype TEXT meegegeven wordt zal er een textveld (textarea) voorzien worden. Bij een VARCHAR(aantal) zal een tekstregel (textfield) voorzien worden, het aantal tekens wordt tevens gevalideerd door het formulier.

Er zijn twee uitzonderingen op deze regel:

1 Indien in de opmerking van een TEXT een SELECT-query terug te vinden is gaat het script ervan uit dat de gegevens uit de query geladen moeten worden. Deze worden omgezet naar een SET. Dit wil zeggen dat meerdere waarden te selecteren zijn.

2 Indien in de opmerking van een VARCHAR() een SELECT-query terug te vinden is gaat het script ervan uit dat de gegevens uit de query geladen moeten worden. Deze worden omgezet naar een ENUM. Dit wil zeggen dat u een keuzelijst krijgt waaruit u één waarde kan selecteren.

Voor beide uitzonderingen zal het script de id's toevoegen indien de query dit voorziet (SELECT id, naam FROM tabel). Indien de query enkel één veld (SELECT naam FROM tabel) gaat toevoegen dan wordt de waarde ingevoegd en geen verwijzing.

### 4 Keuzelijsten en meerkeuzevakjes

Indien het gegevenstype ENUM('optie1','optie2',...) meegegeven wordt zal er een keuzelijst met opties in het formulier weergegeven worden.

Indien het gegevenstype SET('optie1','optie2',...) meegegeven wordt zullen er selectievakjes met de opties in het formulier weergegeven worden. Wanneer meerdere waarden ingevoerd moeten worden dan zal deze array van waarden

omgezet worden naar een string waarin de waarden gescheiden worden door komma's. Deze keuze is gemaakt voor de leesbaarheid van de tabellen te verhogen. Eventueel kan gekozen worden voor serialization in plaats van het imploden en exploden van de array door komma's.

Tot zover hoe het databankontwerp de formulieren zal beïnvloeden. Hieronder kan u het resultaat bekijken.

Create record

<b>Voornaam</b>	<input type="text" value="Bert"/>	
<b>Naam</b>	<input type="text"/>	<b>Dit is een verplicht veld.</b>
<b>Opmerkingen</b>	<div style="border: 1px solid gray; height: 80px;"></div>	
<b>Dienst</b>	<input type="text" value=""/>	<b>Dit is een verplicht veld.</b>
<b>Funcities</b>	<input type="checkbox"/> Administratie <input type="checkbox"/> Coördinator <input type="checkbox"/> Directie	
<b>Geboortedatum</b>	<input type="text"/>	

[Ga terug...](#)

Het formulier wordt opgebouwd door de functie MySQLform(). Deze heeft als parameters de databank, tabel, array van velden en primaire sleutel. Indien een id wordt meegegeven zullen de waarden geladen worden en aangepast bij verzenden. Het is ook mogelijk exceptions toe te voegen aan bovenstaande regels. Misschien wenst u helemaal niet dat het VARCHAR(10) veld paswoord aan te passen is. Daarom kan een array van exeptions meegegeven worden, bijvoorbeeld: \$exceptions["Paswoord"] = "<input type=\"text\" id=\"Paswoord\" text=\"Paswoord\" value=\"Niet aan te passen\" readonly>";

createupdate.php

```
<html>
<head>

<link type="text/css" href="css/site.css" rel="stylesheet" />
<link type="text/css" href="css/table.css" rel="stylesheet" />
<link type="text/css" href="css/smoothness/jquery-ui-1.7.2.custom.css"
rel="stylesheet" />
<script type="text/javascript" src="js/jquery-1.3.2.min.js"></script>
```

```

<script type="text/javascript" src="js/jquery.validate.min.js"></script>

<script type="text/javascript"
src="js/jquery.validate.messages_nl.js"></script>

<script type="text/javascript" src="js/jquery-ui-1.7.2.custom.min.js"></script>

<script type="text/javascript">
$(function() {

    $('input').filter('.datepicker').datepicker({
        dateFormat: 'yy-mm-dd',
        dayNamesMin: ['Zo', 'Ma', 'Di', 'Wo', 'Do', 'Vr',
'Za'],
        firstDay: 1,
        monthNames:
['januari', 'februari', 'maart', 'april', 'mei', 'juni', 'juli', 'august', 'september',
'oktober', 'november', 'december']
    });

});

$(document).ready(function(){
    $("#MySQLform").validate({
        submitHandler: function(form) {
            form.submit();
        }
    });
});
</script>
</head>

<body>

<?php
include_once("MySQLfunctions.php");

include_once("MySQLconnection.php");

// Welke record?
$id=$_GET["id"];

// Update or Create?
if ($id!="") {
    echo "<h1>Update record<h1>";
}
else {
    echo "<h1>Create record<h1>";
}

// Voor sommige velden wens je misschien een specifieke waarde mee te geven...
// Je kan op deze manier ook nog hidden velden gaan toevoegen...
//$exceptions[] = "Paswoord";
//$exceptions["Paswoord"] = "<input type=\"text\" id=\"Paswoord\"
text=\"Paswoord\" value=\"Niet aan te passen\" readonly></input>";

// Calling the MySQLedittable function
$MyFunc->MySQLform($tabel,$fields,$pri_key,$id,$db,$exceptions);

```

```
//Calling the MySQLsisconnect function
$MyFunc->MySQLdisconnect();

echo "<a href=\".\\" target=\"_SELF\">Ga terug...</a>";

?>
</body>
</html>
```

#### Update record

<b>Voornaam</b>	<input type="text" value="Bert"/>
<b>Naam</b>	<input type="text" value="VdA"/>
<b>Opmerkingen</b>	<input type="text" value="Niets speciaals te melden..."/>
<b>Dienst</b>	<input type="text" value="Marketing"/>
<b>Funcies</b>	<input type="checkbox"/> Administratie <input checked="" type="checkbox"/> Coördinator <input checked="" type="checkbox"/> Directie
<b>Geboortedatum</b>	<input type="text" value="2009-11-24"/>

[Ga terug...](#)

Het formulier stuurt zijn waarden door naar replace.php waarde functie MySQLreplace() de INSERT of UPDATE gaat uitvoeren. Deze array van waarden kan aangepast worden. Misschien wenst u een vaste waarde toe te voegen aan uw tabel, bijvoorbeeld:  
 \$waarden["autowaarde"]="Waarden toevoegen aan array van formulier door ".  
 \$waarden["Leerkracht"];

#### replace.php

```
<html>
<head>
<link type="text/css" href="css/site.css" rel="stylesheet" />
</head>
<body>
<h1>Create or Updrate the database via u replace statement.</h1>
<?php
include_once("MySQLfunctions.php");

include_once("MySQLconnection.php");

// Welke waarden?
$waarden=$_POST;

// Wil je zelf nog waarden toevoegen, eventueel op basis van andere velden dan
kan dit hieronder
// Je past de array waarden aan met jouw custom waarden!
//$waarden["autowaarde"]="Waarden toevoegen aan array van formulier door ".
$waarden["Leerkracht"];
```

```
//Calling the replace function
$MyFunc->MySQLreplace($tabel,$waarden);
```



```
//Calling the MySQLsisconnect function
$MyFunc->MySQLdisconnect();

echo "<br/><br/><a href=\".\\" target=\"_SELF\">Ga terug...</a>";

?>
</body>
</html>
```

Tot slot komen we bij de D van CRUD. We wensen de gegevens te deleten, hiervoor zorgt de functie MySQLdelete();

delete.php

```
<html>
<link type="text/css" href="css/site.css" rel="stylesheet" />
<body>
<h1>Delete or inactivate record</h1>
<?php
include_once("MySQLfunctions.php");

include_once("MySQLconnection.php");

// Start validation!!!

// Welke record?
$id=$_GET["id"];
$confirm=$_GET["confirm"];

//Calling the delete function
if ($confirm) {
    $MyFunc->MySQLdelete($tabel,$pri_key,$id);
}
else {
    echo "<center>Aandacht: Deze actie zal de record definitief verwijderen
uit de databank!<br/>";
    echo "<a href=\"?id=$id&confirm=true\">Ja, ik ben zeker!</br></center>";
}

//Calling the MySQLsisconnect function
$MyFunc->MySQLdisconnect();

echo "<br/><br/><a href=\".\\" target=\"_SELF\">Ga terug...</a>";

?>
</body>
</html>
```

Delete or inactivate record

Aandacht: Deze actie zal de record definitief verwijderen uit de databank!  
[Ja, ik ben zeker!](#)

[Ga terug...](#)

Tot zover de mogelijkheden van mijn CRUD functies. Hopelijk brengt het u op ideeën om zelf aan de slag te gaan. Tot slot kijkt u hieronder de eigenlijke functies te zien.

Bert Van den Abbeele

## MySQLfunctions

```
+ MySQLconnect(host : string, user : string, pass : string, db : string) : boolean
+ MySQLtable(db : string, tabel : string, fields : array, pri_key : string, add_delete : boolean, add_edit : boolean, add_new : boolean, where : string, order : string, limit : string)
+ MySQLform(tabel : string, fields : array, pri_key : string, id : int, db : string, exceptions : array)
+ MySQLdelete(tabel : string, pri_key : string, id : int, delete : boolean) : boolean
+ MySQLdisconnect() : boolean
- GetGegevenstypes(tabel : string, key : string) : string
- MySQLrequired(tabel : string, key : string) : boolean
- GetFieldString(fields : array) : string
+ MySQLreplace(tabel : string, waarden : array) : boolean
- GetTypeNoParams(string : string) : string
- GetParamsFromType(string : string) : array
- GetTableCommentFromInformationSchema(db : string, tabel : string, column : string) : string
```

## MySQLfunctions.php

```
<?php

/**
 * class MySQLfunctions
 * Created 2009-09-29 by Bert Van den Abbeele
 *
 */

class MySQLfunctions {

    /**
     * Connecting to MySQL database
     * Created 2009-09-29 by Bert Van den Abbeele
     *
     * @param string $host : MySQL server host
     * @param string $user : username
     * @param string $pass : password
     * @param string $db : database to connect
     *
     * @return boolean
     * @access public
     *
     */

    public function MySQLconnect($host,$user,$pass,$db) {
        $connection = mysql_connect($host, $user, $pass) or die("Could not
connect: " . mysql_error());
        if($connection) {
            $database = mysql_select_db($db,$connection) or die("Could
not select database: " . mysql_error());
            if($database) {
                return true;
            }
            else {
                return false;
            }
        }
        else {
            return false;
        }
    }

    /**
     * Create HTML table from MySQL Query
     * Created 2009-09-22 by Bert Van den Abbeele
     *
     * @param string $db : database
     * @param string $table : tabel
     */
}
```

```

* @param array $fields : array fields and alias
* @param string $pri_key : the key value needed for delete and edit
* @param boolean $add_delete : add delete column
* @param boolean $add_edit : add edit column
* @param string $where
* @param string $order
* @param string $limit
*
* @return void
* @access public
*
*/

public function MySQLtable($db,$tabel,$fields,$pri_key,$add_delete =
false,$add_edit = false, $where = null, $order = null, $limit = null) {

    if($db=="") {
        echo "Error: Geen database ingegeven.";
        die();
    }

    if($tabel=="") {
        echo "Error: Geen tabel ingegeven.";
        die();
    }

    if($fields=="") {
        echo "Error: Geen velden opgegeven.";
        die();
    }

    $fieldstring=$this->GetFieldString($fields);

    // The query is needed to create the table!
    $query="SELECT ";

    if($pri_key!="") {
        $query .= " $pri_key, ";
    }

    $query .= "$fieldstring FROM $tabel";

    if($where!="") {
        $query .= " WHERE $where";
    }

    if($order!="") {
        $query .= " ORDER BY $order";
    }

    if($limit!="") {
        $query .= " LIMIT $limit";
    }

    if($query=="") {
        echo "Error: No SQL query found.";
        die();
    }

    // Build records table
    echo "<table name=\"MySQLtable\" id=\"MySQLtable\" width=\"100%\"
cellspacing=\"0\" cellpadding=\"1\" border=\"1\">\n";

```

```

        $result = mysql_query($query) or die("Could not query data: " .
mysql_error());

        $aantal=mysql_num_rows($result);

        // Adding head
        echo "<tr>\n";
        foreach($fields as $alias => $field) {
            if($alias!=$pri_key) {
                echo "<th>" . $alias . "</th>\n";
            }
        }
        if($add_edit===true && $pri_key!=""){
            echo "<th></th>";
        }
        if($add_delete===true && $pri_key!=""){
            echo "<th></th>";
        }
        if($add_edit===false && $add_delete===false && $add_new===true &&
$pri_key!=""){
            echo "<th></th>";
        }
        echo "</tr><tr>\n";

        // Adding records
        while ($row = mysql_fetch_array($result, MYSQL_ASSOC)) {

            foreach($row as $alias => $value) {
                $row[$alias] = stripslashes($value);
            }

            foreach($row as $alias => $value) {
                if($alias!=$pri_key) {
                    // De opmerking bekijken of er een SELECT
statement in staat
                    $opmerking=$this-
>GetTableCommentFromInformationSchema($db,$tabel,$fields[$alias]);
                    if (substr($opmerking,0,6)=="SELECT") {
                        // Select gevonden in opmerking
                        $query2=$opmerking;
                        $result2 = mysql_query($query2) or
die("Could not query data: " . mysql_error());
                        $params=array();
                        $aantalcolumns=mysql_num_fields($result2);
                        $columnsnaam =
mysql_field_name($result2,0);
                        if ($aantalcolumns==2) {
                            // Waarden opzoeken vanuit id!
                            // Array opbouwen van de set (gewoon
komma's ertussen ipv serialized)
                            $array=explode(",",$value);
                            $inhoud=array();
                            foreach ($array as $waarde) {
                                $query3="$opmerking WHERE
$columnsnaam=$waarde";
                                $result3 = mysql_query($query3)
or die("Could not query data: " . mysql_error());
                                $row3 =
mysql_fetch_array($result3);
                                $inhoud[] = $row3[1];
                            }

```

```

                                echo "<td>".implode(", ",
$inhoud)."</td>";
                                }
                                else {
de waarde werd ingegeven                                // Waarden niet opzoeken, geen id dus
                                echo "<td>" . nl2br($value) .
"</td>\n";
                                }
                                }
                                else {
                                echo "<td>" . nl2br($value) . "</td>\n";
                                }
                                }
                                }
                                if($add_edit===true && $pri_key!=""){
                                echo "<td width=\"25\"><a href=\"createupdate.php?
id=$row[$pri_key]\" targer=\"_self\"><img src=\"images/edit.png\"
border=\"0\" /></a></td>";
                                }
                                if($add_delete===true && $pri_key!=""){
                                echo "<td width=\"25\"><a href=\"delete.php?
id=$row[$pri_key]\" targer=\"_self\"><img src=\"images/delete.png\"
border=\"0\" /></a></td>";
                                }
                                if($add_edit===false && $add_delete===false &&
$add_new===true && $pri_key!=""){
                                echo "<td></td>";
                                }
                                }
                                echo "</tr><tr>\n";
                                }

                                echo "</table>\n";

                                mysql_free_result($result);

                                echo "</table>\n";

                                }

/**
 * Make table to edit values from MySQL table
 * Created 2009-10-02 by Bert Van den Abbeele
 *
 * @param string $table : tabel
 * @param array $fields : array fields and alias
 * @param string $pri_key : the key of table
 * @param int $id : record to edit
 * @param string $db : database
 * @param array $exceptions : fields to add or modify on form generation
 *
 * @return void
 * @access public
 *
 */

public function MySQLform($tabel,$fields,$pri_key,$id,$db,
$exceptions=array()) {

    if($tabel=="") {
        echo "Error: Geen tabel ingegeven.";
        die();
    }

```

```

    }

    if($pri_key=="") {
        echo "Error: Geen pri_key opgegeven.";
        die();
    }

    if($fields=="") {
        echo "Error: Geen velden opgegeven.";
        die();
    }

    $fieldstring=$this->GetFieldString($fields);

    if($id=="") {
        // Creating a new record! So the key needs to be null!
        $id="null";
    }

    // The query is needed to create te table!
    $query="SELECT $fieldstring FROM $tabel WHERE $pri_key = $id";

    $result = mysql_query($query) or die("Could not query data: " .
mysql_error());

    $row = mysql_fetch_array($result, MYSQL_ASSOC);

    $r=0;

    echo "<form id=\"MySQLform\" name=\"MySQLform\" method=\"post\"
action=\"replace.php\">\n";
    echo "<fieldset>\n";
    echo "<input type=\"hidden\" name=\"id\" id=\"id\"
value=\"\$id\">\n";
    foreach($fields as $alias => $field) {
        $type=$this->GetGegevenstypes($tabel,$field);
        $required=$this->MySQLrequired($tabel,$field);
        //echo "<label>" . $alias .
"(".substr($type,0,20).")</label>\n";
        echo "<label>" . $alias . "</label>\n";

        $typenoparams=$this->GetTypeNoParams($type);

        // De opmerking bekijken of er een SELECT statement in staat
        $opmerking=$this->GetTableCommentFromInformationSchema($db,
$tabel,$field);
        if (substr($opmerking,0,6)=="SELECT") {
            $typenoparams="select";
        }

        // Hier testen we over een een "user defined exception" is
        voor dit veld. De echte veldnaam wordt doorgegeven!
        if (in_array($field,$exceptions)) {
            $typenoparams="custom";
        }

        switch ($typenoparams) {
            case "text":
                echo "<textarea name=\"".$field."\" id=\"".$field."\"
rows=\"6\" cols=\"40\" ";
                // Is NotNull then required (validation)
                if($required) {

```

```

        echo " class=\"required\"";
    }
    echo
">".stripslashes($row[$alias])."</textarea><br/>\n";
    break;
    case "date":
        echo "<input type=\"text\" name=\"".$field."\" id=\"".$field."\" value=\"".stripslashes($row[$alias]).\"\" size=\"20\" class=\"
datepicker dateISO";
        // Is NotNull then required (validation)
        if($required) {
            echo " required\"";
        }
        else {
            echo "\"";
        }
        echo " /><br/>\n";
        break;
    case "varchar":
        $maxlength=$this->GetParamsFromType($type);
        echo "<input type=\"text\" name=\"".$field."\" id=\"".$field."\" value=\"".$row[$alias].\"\" size=\"".$maxlength[0].\"\" maxlength=\"".$maxlength[0].\"\"";
        // Is NotNull then required (validation)
        if($required) {
            echo " class=\"required\"";
        }
        echo " /><br/>\n";
        break;
    case "enum":
        $params=$this->GetParamsFromType($type);

        echo "<select name=\"".$field."\" id=\"".$field."\"";
        // Is NotNull then required (validation)
        if($required) {
            echo " class=\"required\"";
        }
        echo " /><br/>\n";
        echo "<option value=\"\"></option>\n";
        // Hier de mogelijkheden!
        foreach($params as $option) {
            echo "<option ";
            if ($option==stripslashes($row[$alias])) {
                echo "selected";
            }
            echo " value=\"".$option.">".$option</option>\n";
        }
        echo "</select><br/>\n";

    break;
    case "set":
        $params=$this->GetParamsFromType($type);

        //array opbouwen van de set (gewoon komma's ertussen
ipv serialized)
        $array=explode(",",$row[$alias]);

        echo "<table border=\"0\">\n";
        $i=0;
        foreach($params as $option) {
            echo "<tr><td><input ";
            if (in_array($option,$array)) {

```

```

        echo "checked ";
    }
    echo "type=checkbox name=\"\".$field.\"[]\" id=\"\".
$field.\"[]\" value=\"\$option\"";
    if($required) {
        echo " class=\"required\"";
    }
    echo " />$option </td></tr>\n";
}
echo "</table>\n";
break;
case "select":
    // Select gevonden in opmerking
    $query2=$opmerking;
    $result2 = mysql_query($query2) or die("Could not query
data: " . mysql_error());
    $params=array();
    // Indien SELECT id, waarde FROM... Gebruiken we de id!
    $aantalcolumns=mysql_num_fields($result2);
    if ($aantalcolumns==2) {
        while($row2 = mysql_fetch_array($result2)) {

            $params[$row2[0]] = $row2[1];
        }
    }
    else {
        while($row2 = mysql_fetch_array($result2)) {

            //array_unshift($params,$row2[0]);
            $params[$row2[0]] = $row2[0];
        }
    }

    if($type=="text") {
        // Type text met een select als opmerking geeft
een set

//array opbouwen van de set (gewoon komma's
ertussen ipv serialized)

$array=explode(",",$row[$alias]);

echo "<table border=\"0\">\n";
$i=0;
foreach($params as $option => $value) {
    echo "<tr><td><input ";
    if (in_array($option,$array)) {
        echo "checked ";
    }
    echo "type=checkbox name=\"\".$field.\"[]\"
id=\"\".$field.\"[]\" value=\"\$option\"";
    if($required) {
        echo " class=\"required\"";
    }
    echo " />$value</td></tr>\n";
}
echo "</table>\n";
}
elseif ($this->GetTypeNoParams($type)=="varchar") {
    // Type varchar met een select als opmerking
heeft een enum

echo "<select name=\"\".$field.\"\" id=\"\".
$field.\"\"";

```



```

        // Is NotNull then required (validation)

        if($required) {
            echo " class=\"required\"";
        }
        echo " /><br/>\n";
        echo "<option value=\"\"></option>\n";
        // Hier de mogelijkheden!
        foreach($params as $option => $value) {
            echo "<option ";
            if ($option==stripslashes($row[$alias])) {
                echo "selected";
            }
            echo "
value=\"\$option\">$value</option>\n";
        }
        echo "</select><br/>\n";
    }
    else {
        echo "Select in opmerking maar geen varchar of
text!";
    }
    break;
    case "custom":
        echo "$exceptions[$field]<br/>\n";
        break;
    default:
        echo "Gegevenstype niet ondersteund!<br/>\n";
    }
    $r++;
}
echo "</fieldset>\n";
echo "<center><input type=\"submit\" value=\"Aanpassen\"
/></center></td>\n";

    echo "</form>\n";

    mysql_free_result($result);
}

/**
 * Delete of inactivate a record
 * Created 2009-10-02 by Bert Van den Abbeele
 *
 * @param string $tabel : tabel
 * @param string $pri_key : the key of the table
 * @param int $id : record to delete
 * @param boolean $delete : delete of inactivate record
 *
 * @return boolean
 * @access public
 */
public function MySQLdelete($tabel,$pri_key,$id,$delete=false) {
    if($tabel=="") {
        echo "Error: No tabel found.";
        die();
    }
    if($id=="") {

```

```

        echo "Error: Nothing to delete or inactivate!";
        die();
    }

    $query="DELETE FROM $tabel WHERE $pri_key=$id LIMIT 1\n";

    $result = mysql_query($query) or die("Could not query data: " .
mysql_error());

    //echo "Met succes verwijderd!";

    if($result) {
        return true;
    }
    else {
        return false;
    }
}

/**
 * Disconnecting from MySQL database
 * Created 2009-09-29 by Bert Van den Abbeele
 *
 * @return boolean
 * @access public
 */

public function MySQLdisconnect() {
    $disconnect=mysql_close() or die("Could not disconnect: " .
mysql_error());
    if($disconnect) {
        return true;
    }
    else {
        return false;
    }
}

/**
 * GetGegevenstypes to Array
 * Created 2009-10-02 by Bert Van den Abbeele
 *
 * @param string $tabel : tabel
 * @param string $key : te controleren veld
 *
 * @return string
 * @access private
 */

private function GetGegevenstypes($tabel,$key) {

    if($tabel=="") {
        echo "Error: Geen tabel ingegeven.";
        die();
    }

    $query="DESCRIBE $tabel $key";

    $result = mysql_query($query) or die("Could not query data: " .

```

```

mysql_error());

    $row = mysql_fetch_array($result, MYSQL_ASSOC);

    return $row[Type];

}

/**
 * Null allowed? Indien niet dan add class="required"
 * Created 2009-10-02 by Bert Van den Abbeele
 *
 * @param string $tabel : tabel
 * @param string $key : te controleren veld
 *
 * @return boolean
 * @access private
 */
private function MySQLrequired($tabel,$key) {

    if($tabel=="") {
        echo "Error: Geen tabel ingegeven.";
        die();
    }

    $query="DESCRIBE $tabel $key";

mysql_error());

    $result = mysql_query($query) or die("Could not query data: " .
mysql_error());

    $row = mysql_fetch_array($result, MYSQL_ASSOC);

    if($row["Null"]=="NO") {
        return true;
    }
    else {
        return false;
    }

}

/**
 * Array[field => alias] to String
 * Created 2009-10-02 by Bert Van den Abbeele
 *
 * @param array fields
 *
 * @return string
 * @access private
 */
private function GetFieldString($fields) {
    //Making the fields string
    $fieldsaantal = count($fields);
    $f=0;
    foreach($fields as $alias => $field) {
        $fieldstring .= $field;
        if($alias) {
            $fieldstring .= " AS ".$alias;
        }
        $f++;
    }
}

```

```

        if($f!=$fieldsaantal) {
            $fieldstring .= ",";
        }
    }
    return $fieldstring;
}

/**
 * Replace functie
 * Created 2009-10-02 by Bert Van den Abbeele
 *
 * @param string $tabel : tabel
 * @param array $waarden : array van waarden (afkomstig van het formulier
$_POST)
 *
 * @return boolean
 * @access public
 */

public function MySQLreplace($tabel,$waarden) {

    // Validatie
    foreach($waarden AS $field => $value) {
        // Als waarde array is dan aanpassen
        if (is_array($value)) {
            $value=implode(",",$value);
        }
        $waarden[$field] =
htmlspecialchars(mysql_real_escape_string($value));
        //$waarden[$field] = mysql_real_escape_string($value);

    }

    $query="REPLACE INTO $tabel SET ";

    $aantalrecords=count($waarden);
    $i=1;

    foreach($waarden as $field => $value) {
        $query.=" $field='$value'";
        if($i!=$aantalrecords) {
            $query.=",";
        }
        $i++;
    }

    $result = mysql_query($query) or die("Could not query data: " .
mysql_error());

    //echo "Met succes uitgevoerd!";

    if($result) {
        return true;
    }
    else {
        return false;
    }

}

```

```

/**
 * Get type without parameters
 * Created 2009-10-04 by Bert Van den Abbeele
 *
 * @param string $string : the string
 *
 * @return string
 * @access private
 */
private function GetTypeNoParams($string) {
    $position=strpos($string,"(",0) ;
    if($position) {
        return substr($string,0,$position);
    }
    else {
        return $string;
    }
}

/**
 * Get parameters from type to array
 * Created 2009-10-04 by Bert Van den Abbeele
 *
 * @param string $string : the string
 *
 * @return array
 * @access private
 */
private function GetParamsFromType($string) {
    $start=strpos($string,"(",0) ;
    $stop=strpos($string,")",0) ;
    $string=substr($string,$start+1,$stop-($start+1));
    $array=explode(",",$string);
    foreach ($array as $key => $value) {
        $array[$key]=trim($value,"\x22\x27");
    }
    return $array;
}

/**
 * Opmerkingen uit de information schema halen
 * Created 2009-10-04 by Bert Van den Abbeele
 *
 * @param string $db
 * @param string $tabel
 * @param string $column
 *
 * @return string
 * @access private
 */
private function GetTableCommentFromInformationSchema($db,$tabel,$column)
{
    if($db=="" || $tabel =="" || $column=="") {
        return "";
    }
}

```

```
    }  
    $query="SELECT COLUMN_COMMENT FROM INFORMATION_SCHEMA.`COLUMNS`  
WHERE TABLE_SCHEMA='$db' AND TABLE_NAME='$tabel' AND COLUMN_NAME='$column';  
    $result = mysql_query($query) or die("Could not query data: " .  
mysql_error());  
    $row = mysql_fetch_array($result, MYSQL_ASSOC);  
    return $row["COLUMN_COMMENT"];  
    }  
}  
?>
```